CHAPTER

# 1

# Introduction to Design Optimization

Upon completion of this chapter, you will be able to:

- Describe the overall process of designing systems
- Distinguish between engineering design and engineering analysis activities
- Distinguish between the conventional design process and optimum design process
- Distinguish between optimum design and optimal control problems
- Understand the notations used for operations with vectors, matrices, and functions and their derivatives

Engineering consists of a number of well-established activities, including analysis, design, fabrication, sales, research, and development of systems. The subject of this text—the design of systems—is a major field in the engineering profession. The process of designing and fabricating systems has been developed over centuries. The existence of many complex systems, such as buildings, bridges, highways, automobiles, airplanes, space vehicles, and others, is an excellent testimonial to its long history. However, the evolution of such systems has been slow and the entire process is both time-consuming and costly, requiring substantial human and material resources. Therefore, the procedure is to design, fabricate, and use a system regardless of whether it is the *best one.* Improved systems have been designed only after a substantial investment has been recovered.

The preceding discussion indicates that several systems can usually accomplish the same task, and that some systems are better than others. For example, the purpose of a bridge is to provide continuity in traffic from one side of the river to the other. Several types of bridges can serve this purpose. However, to analyze and design all possibilities can be time-consuming and costly. Usually one type is selected based on some preliminary analyses and is designed in detail.

The design of a system can be *formulated as a problem of optimization* in which a performance measure is optimized while all other requirements are satisfied. Many numerical methods of optimization have been developed and used to design better systems. This text describes the basic concepts of optimization and numerical methods for the design of engineering systems. Design process, rather than optimization theory, is emphasized. Various theorems are stated

**Introduction to Optimum Design.** http://dx.doi.org/10.1016/B978-0-12-800806-5.00001-9

as results without rigorous proofs. However, their implications from an engineering point of view are discussed.

Any problem in which certain parameters need to be determined to satisfy constraints can be formulated as an optimization problem. Once this has been done, optimization concepts and methods described in this text can be used to solve it. Optimization methods are quite general, having a wide range of applicability in diverse fields. It is not possible to discuss every application of optimization concepts and methods in this introductory text. However, using simple applications, we discuss concepts, fundamental principles, and basic techniques that are used in most applications. The student should understand them without getting bogged down with notations, terminologies, and details on particular areas of application.

## 1.1 THE DESIGN PROCESS

### How Do I Begin to Design a System?

Designing engineering systems can be a complex process. Assumptions must be made to develop realistic models that can be subjected to mathematical analysis by the available methods. The models may need to be verified by experiments. Many possibilities and factors must be considered during the optimization problem formulation phase. *Economic considerations* play an important role in designing cost-effective systems. To complete the design of an engineering system, designers from different fields of engineering must usually cooperate. For example, the design of a high-rise building involves designers from architectural, structural, mechanical, electrical, and environmental engineering, as well as construction management experts. Design of a passenger car requires cooperation among structural, mechanical, automotive, electrical, chemical, hydraulics design, and human factor engineers. Thus, in an *interdisciplinary environment*, considerable interaction is needed among design teams to complete the project. For most applications, the entire design project must be broken down into several subproblems, which are then treated somewhat independently. Each of the subproblems can be posed as a problem of optimum design.

The design of a system begins with the analysis of various options. Subsystems and their components are identified, designed, and tested. This process results in a set of drawings, calculations, and reports with the help of which the system can be fabricated. We use a systems engineering model to describe the *design process.* Although complete discussion of this subject is beyond the scope of this text, some basic concepts are discussed using a simple block diagram.

Design is an *iterative process*. *Iterative* implies analyzing several *trial designs* one after another until an acceptable design is obtained. It is important to understand the concept of a trial design. In the design process, the designer estimates a trial design of the system based on experience, intuition, or some simple mathematical analyses. The trial design is then analyzed to determine if it is acceptable. In case it gets accepted, the design process is terminated. In the optimization process, the trial design is analyzed to determine if it is the best. Depending on the specifications, "best" can have different connotations for different systems. In general, it implies that a system is cost-effective, efficient, reliable, and durable.

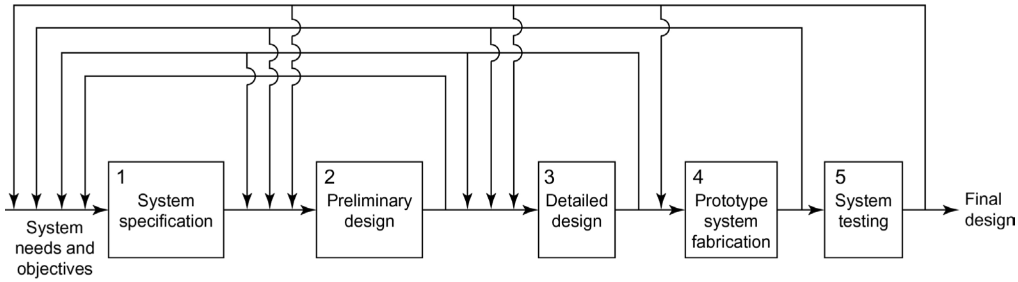


**FIGURE 1.1 System evolution model.**

The basic concepts are described in this text to aid the engineer in designing systems at minimum cost.

The design process should be well organized. To discuss it, we consider a *system evolution model*, shown in Fig. 1.1, where the process begins with the identification of a need that may be conceived by engineers or nonengineers. The five steps of the model in the figure are described in the following paragraphs.

1. The *first step* in the evolutionary process is to precisely define the specifications for the system. Considerable interaction between the engineer and the sponsor of the project is usually necessary to quantify the *system specifications.*
2. The *second step* in the process is to develop a *preliminary design* of the system. Various system concepts are studied. Since this must be done in a relatively short time, *simplified models* are used at this stage. Various subsystems are identified and their preliminary designs are estimated. Decisions made at this stage generally influence the system's final appearance and performance. At the end of the preliminary design phase, a few promising design concepts that need further analysis are identified.
3. The *third step* in the process is a *detailed design* for all subsystems using the iterative process described earlier. To evaluate various possibilities, this must be done for all previously identified promising design concepts. The design parameters for the subsystems must be identified. The system performance requirements must be identified and formulated. The subsystems must be designed to maximize system worth or to minimize a measure of the cost. Systematic optimization methods described in this text aid the designer in accelerating the detailed design process. At the end of the process, a description of the final design is available in the form of reports and drawings.
4. The *fourth and fifth steps* shown in Fig. 1.1 may or may not be necessary for all systems. They involve fabrication of a prototype system and testing, and are necessary when the system must be mass-produced or when human lives are involved. These steps may appear to be the final ones in the design process, but they are not because the system may not perform according to specifications during the testing phase. Therefore, the specifications may have to be modified or other concepts may have to be studied. In fact, this reexamination may be necessary at any point during the design process. It is for this reason that *feedback loops* are placed at every stage of the system evolution process, as shown in Fig. 1.1. This iterative process must be continued until the best system evolves.

Depending on the complexity of the system, this process may take a few days or several months.

The model described in Fig. 1.1 is a simplified block diagram for system evolution. In actual practice, each block may be broken down into several subblocks to carry out the studies properly and arrive at rational decisions. *The important point is that optimization concepts and methods are helpful at every stage of the process.* Such methods, along with the appropriate software, can be useful in studying various design possibilities rapidly.

## 1.2 ENGINEERING DESIGN VERSUS ENGINEERING ANALYSIS

### Can I Design Without Analysis?

#### ***No, You Must Analyze!***

It is important to recognize the differences between *engineering analysis* and *design activities*. The analysis problem is concerned with determining the behavior of an existing system or a trial system being designed for a known task. Determination of the behavior of the system implies calculation of its response to specified inputs. For this reason, the sizes of various parts and their configurations are given for the analysis problem; that is, the design of the system is known. On the other hand, the design process calculates the sizes and shapes of various parts of the system to meet performance requirements.

The design of a system is an *iterative process*. We estimate a trial design and analyze it to see if it performs according to given specifications. If it does, we have an *acceptable* (*feasible*) *design*, although we may still want to change it to improve its performance. If the trial design does not work, we need to change it to come up with an acceptable system. In both cases, we must be able to *analyze designs* to make further decisions. Thus, analysis capability must be available in the design process.

This book is intended for use in all branches of engineering. It is assumed throughout that students understand the analysis methods covered in undergraduate engineering statics and physics courses. However, *we will not let the lack of analysis capability hinder understanding of the systematic process of optimum design*. Equations for analysis of the system are given wherever feasible.

## 1.3 CONVENTIONAL VERSUS OPTIMUM DESIGN PROCESS

### Why Do I Want to Optimize?

#### ***Because You Want to Beat the Competition and Improve Your Bottom Line!***

It is a challenge for engineers to design efficient and cost-effective systems without compromising their integrity. Fig. 1.2a presents a self-explanatory flowchart for a conventional design method; Fig. 1.2b presents a similar flowchart for the optimum design method. It is important to note that both methods are iterative, as indicated by a loop between blocks 6 and 3. Both methods have some blocks that require similar calculations and others that require different calculations. The key features of the two processes are as follows.

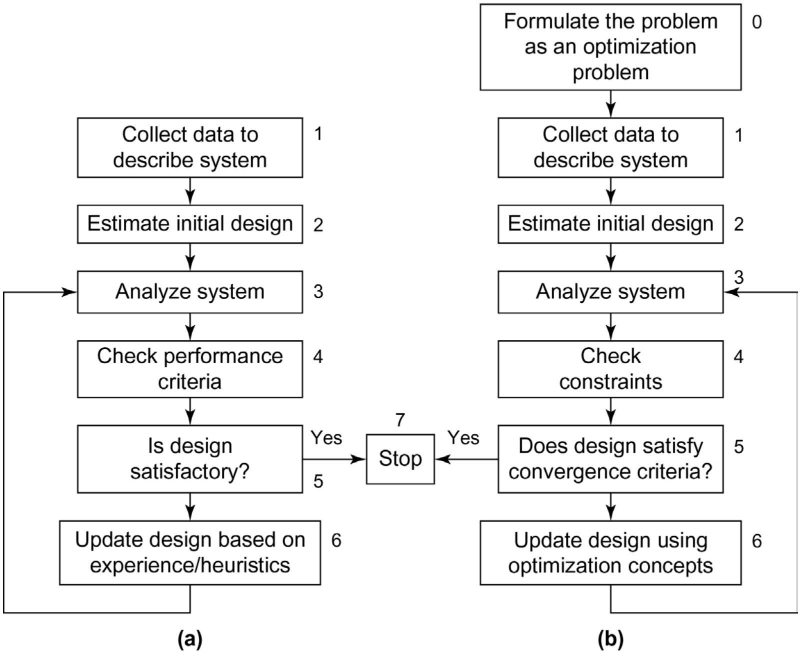


**FIGURE 1.2** Comparison of: (a) conventional design method; and (b) optimum design method.

**0.** The optimum design method has block 0, where the problem is formulated as one of optimization (see chapter: Optimum Design Problem Formulation for detailed discussion). An objective function is defined that measures the merits of different designs.
**1.** Both methods require data to describe the system in block 1.
**2.** Both methods require an initial design estimate in block 2.
**3.** Both methods require analysis of the system in block 3.
**4.** In block 4, the conventional design method checks to ensure that the performance criteria are met, whereas the optimum design method checks for satisfaction of all of the constraints for the problem formulated in block 0.
**5.** In block 5, stopping criteria for the two methods are checked, and the iteration is stopped if the specified stopping criteria are met.
**6.** In block 6, the conventional design method updates the design based on the designer's experience and intuition and other information gathered from one or more trial designs; the optimum design method uses optimization concepts and procedures to update the current design.

The foregoing distinction between the two design approaches indicates that the conventional design process is less formal. An objective function that measures a design's merit is not identified. Trend information is usually not calculated; nor is it used in block 6 to make design decisions for system improvement. In contrast, the optimization process is more formal, using trend information to make design changes.

## 1.4 OPTIMUM DESIGN VERSUS OPTIMAL CONTROL

### What Is Optimal Control?

Optimum design and optimal control of systems are separate activities. There are numerous applications in which methods of optimum design are useful in designing systems. There are many other applications where optimal control concepts are needed. In addition, there are some applications in which both optimum design and optimal control concepts must be used. Sample applications of both techniques include *robotics* and *aerospace structures.* In this text, optimal control problems and methods are not described in detail. However, the fundamental differences between the two activities are briefly explained in the sequel. It turns out that some optimal control problems can be transformed into optimum design problems and treated by the methods described in this text. Thus, methods of optimum design are very powerful and should be clearly understood. A simple optimal control problem is described in chapter: Practical Applications of Optimization and is solved by the methods of optimum design.

The optimal control problem consists of finding feedback controllers for a system to produce the desired output. The system has active elements that sense output fluctuations. System controls are automatically adjusted to correct the situation and optimize a measure of performance. Thus, control problems are usually dynamic in nature. In optimum design, on the other hand, we design the system and its elements to optimize an objective function. The system then remains fixed for its entire life.

As an example, consider the cruise control mechanism in passenger cars. The idea behind this feedback system is to control fuel injection to maintain a constant speed. Thus, the system's output (ie, the vehicle's cruising speed) is known. The job of the control mechanism is to sense fluctuations in speed depending on road conditions and to adjust fuel injection accordingly.

## 1.5 BASIC TERMINOLOGY AND NOTATION

### Which Notation Do I Need to Know?

To understand and to be comfortable with the methods of optimum design, a student must be familiar with linear algebra (vector and matrix operations) and basic calculus. Operations of *linear algebra* are described in Appendix A. Students who are not comfortable with this material need to review it thoroughly. Calculus of functions of single and multiple variables must also be understood. Calculus concepts are reviewed wherever they are needed. In this section, the *standard terminology* and *notations* used throughout the text are defined. It is important to understand and memorize these notations and operations.

### 1.5.1 Vectors and Points

Since realistic systems generally involve several variables, it is necessary to define and use some convenient and compact notations to represent them. Set and vector notations serve this purpose quite well.

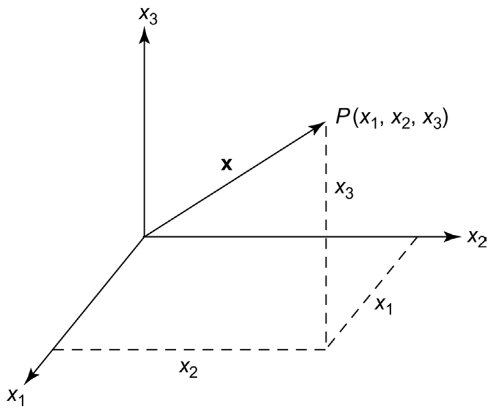


FIGURE 1.3 **Vector representation of a point *P* in 3D space.**

*A point is an ordered list of numbers*. Thus, $(x_1, x_2)$ is a point consisting of two numbers whereas $(x_1, x_2, \ldots, x_n)$ is a point consisting of $n$ numbers. Such a point is often called an $n$-tuple. The $n$ components $x_1, x_2, \ldots, x_n$ are collected into a column vector as

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} x_1 \; x_2 \ldots x_n \end{bmatrix}^T \tag{1.1}$$

where the superscript $T$ denotes the *transpose* of a vector or a matrix. This is called an *n-vector*. Each number $x_i$ is called a component of the (point) vector. Thus, $x_1$ is the first component, $x_2$ is the second, and so on.

We also use the following notation to represent a point or a vector in the $n$-dimensional space:

$$\mathbf{x} = (x_1, x_2, \ldots, x_n) \tag{1.2}$$

In 3-dimensional (3D) space, the vector $\mathbf{x} = [x_1 \; x_2 \; x_3]^T$ represents a point $P$, as shown in Fig. 1.3. Similarly, when there are $n$ components in a vector, as in Eqs. (1.1) and (1.2), $\mathbf{x}$ is interpreted as a point in the $n$-dimensional space, denoted as $R^n$. The space $R^n$ is simply the collection of all $n$-dimensional vectors (points) of real numbers. For example, the real line is $R^1$, the plane is $R^2$, and so on.

**The terms *vector* and *point* are used interchangeably, and lowercase letters in roman boldface are used to denote them. Uppercase letters in roman boldface represent matrices.**

### 1.5.2 Sets

Often we deal with *sets* of points satisfying certain conditions. For example, we may consider a set $S$ of all points having three components, with the last having a fixed value of 3, which is written as

$$S = \{\mathbf{x} = (x_1, x_2, x_3) \mid x_3 = 3\} \tag{1.3}$$

Information about the set is contained in braces ({ }). Eq. (1.3) reads as "$S$ equals the set of all points ($x_1$, $x_2$, $x_3$) with $x_3 = 3$." The vertical bar divides information about the set $S$ into two parts: To the left of the bar is the dimension of points in the set; to the right are the properties that distinguish those points from others not in the set (eg, properties a point must possess to be in the set $S$).

Members of a set are sometimes called *elements*. If a point $\mathbf{x}$ is an element of the set $S$, then we write $\mathbf{x} \in S$. The expression $\mathbf{x} \in S$ is read as "$\mathbf{x}$ is an element of (belongs to) $S$." Conversely, the expression "$\mathbf{y} \notin S$" is read as "$\mathbf{y}$ is not an element of (does not belong to) $S$."

If all the elements of a set $S$ are also elements of another set $T$, then $S$ is said to be a *subset* of $T$. Symbolically, we write $S \subset T$, which is read as "$S$ is a subset of $T$" or "$S$ is contained in $T$." Alternatively, we say "$T$ is a superset of $S$," which is written as $T \supset S$.

As an example of a set $S$, consider a domain of the $x_1 - x_2$ plane enclosed by a circle of radius 3 with the center at the point (4, 4), as shown in Fig. 1.4. Mathematically, all points within and on the circle can be expressed as

$$S = \{\mathbf{x} \in R^2 \mid (x_1 - 4)^2 + (x_2 - 4)^2 \le 9\} \tag{1.4}$$

Thus, the center of the circle (4, 4) is in the set $S$ because it satisfies the inequality in Eq. (1.4). We write this as $(4, 4) \in S$. The origin of coordinates (0, 0) does not belong to the set because it does not satisfy the inequality in Eq. (1.4). We write this as $(0, 0) \notin S$. It can be verified that the following points belong to the set: (3, 3), (2, 2), (3, 2), (6, 6). In fact, set $S$ has an infinite number of points. Many other points are not in the set. It can be verified that the following points are not in the set: (1, 1), (8, 8), and (−1, 2).

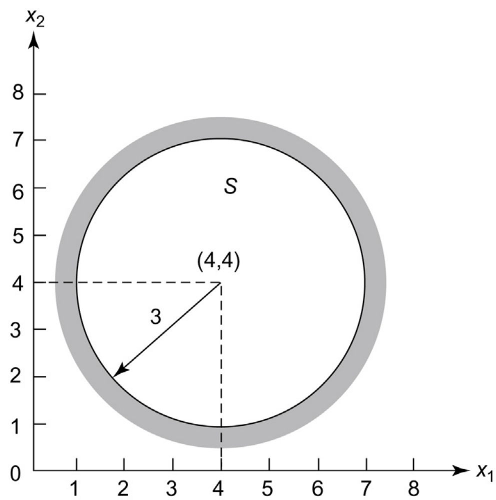


**FIGURE 1.4 Geometrical representation of the set** $S = \{\mathbf{x} \mid (x_1 - 4)^2 + (x_2 - 4)^2 \le 9\}$.

### 1.5.3 Notation for Constraints

Constraints arise naturally in optimum design problems. For example, the material of the system must not fail, the demand must be met, resources must not be exceeded, and so on. We shall discuss the constraints in more detail in chapter: Optimum Design Problem Formulation. Here we discuss the terminology and notations for the constraints.

We encountered a constraint in Fig. 1.4 that shows a set $S$ of points within and on the circle of radius 3. The set $S$ is defined by the following constraint:

$$(x_1 - 4)^2 + (x_2 - 4)^2 \le 9 \tag{1.5}$$

A constraint of this form is a "less than or equal to type" *constraint* and is abbreviated as "≤ type." Similarly, there are *greater than or equal to type constraints*, abbreviated as "≥ type." Both are called *inequality constraints*.

### 1.5.4 Superscripts/Subscripts and Summation Notation

Later we will discuss a set of vectors, components of vectors, and multiplication of matrices and vectors. To write such quantities in a convenient form, consistent and compact notations must be used. We define these notations here. *Superscripts are used to represent different vectors and matrices.* For example, $\mathbf{x}^{(i)}$ represents the $i$th vector of a set and $\mathbf{A}^{(k)}$ represents the $k$th matrix. *Subscripts are used to represent components of vectors and matrices.* For example, $x_j$ is the $j$th component of $\mathbf{x}$ and $a_{ij}$ is the $i$–$j$th element of matrix $\mathbf{A}$. Double subscripts are used to denote elements of a matrix.

To indicate the *range of a subscript or superscript* we use the notation

$$x_i;\ i = 1 \text{ to } n \tag{1.6}$$

This represents the numbers $x_1, x_2, \ldots, x_n$. Note that "$i = 1$ to $n$" represents the range for the index $i$ and is read, "$i$ goes from 1 to $n$." Similarly, a set of $k$ vectors, each having $n$ components, is represented by the *superscript* notation as

$$\mathbf{x}^{(j)};\ j = 1 \text{ to } k \tag{1.7}$$

This represents the $k$ vectors $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \ldots, \mathbf{x}^{(k)}$. It is important to note that subscript $i$ in Eq. (1.6) and superscript $j$ in Eq. (1.7) are *free indices*; that is, they can be replaced by any other variable. For example, Eq. (1.6) can also be written as $x_j$, $j = 1$ to $n$ and Eq. (1.7) can be written as $\mathbf{x}^{(i)}$, $i = 1$ to $k$. Note that the superscript $j$ in Eq. (1.7) does not represent the power of $\mathbf{x}$. It is an index that represents the $j$th vector of a set of vectors.

We also use the *summation notation* quite frequently. For example,

$$c = x_1y_1 + x_2y_2 + \ldots + x_ny_n \tag{1.8}$$

is written as

$$c = \sum_{i=1}^{n} x_iy_i \tag{1.9}$$

Also, multiplication of an $n$-dimensional vector $\mathbf{x}$ by an $m \times n$ matrix $\mathbf{A}$ to obtain an $m$-dimensional vector $\mathbf{y}$ is written as

$$\mathbf{y} = \mathbf{A}\mathbf{x} \tag{1.10}$$

Or, in summation notation, the $i$th component of $\mathbf{y}$ is

$$y_i = \sum_{j=1}^{n} a_{ij} x_j = a_{i1} x_1 + a_{i2} x_2 + \ldots + a_{in} x_n; \quad i = 1 \text{ to } m \tag{1.11}$$

There is another way of writing the matrix multiplication of Eq. (1.10). Let $m$-dimensional vectors $\mathbf{a}^{(i)}$; $i = 1$ to $n$ represent columns of the matrix $\mathbf{A}$. Then $\mathbf{y} = \mathbf{A}\mathbf{x}$ is also written as

$$\mathbf{y} = \sum_{j=1}^{n} \mathbf{a}^{(j)} x_j = \mathbf{a}^{(1)} x_1 + \mathbf{a}^{(2)} x_2 + \ldots + \mathbf{a}^{(n)} x_n \tag{1.12}$$

The sum on the right side of Eq. (1.12) is said to be a *linear combination* of columns of matrix $\mathbf{A}$ with $x_j$, $j = 1$ to $n$ as its multipliers. Or $\mathbf{y}$ is given as a linear combination of columns of $\mathbf{A}$ (refer Appendix A for further discussion of the linear combination of vectors).

Occasionally, we must use the double summation notation. For example, assuming $m = n$ and substituting $y_i$ from Eq. (1.11) into Eq. (1.9), we obtain the double sum as

$$c = \sum_{i=1}^{n} x_i \left( \sum_{j=1}^{n} a_{ij} x_j \right) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j \tag{1.13}$$

Note that the indices $i$ and $j$ in Eq. (1.13) can be interchanged. This is possible because $c$ is a *scalar quantity*, so its value is not affected by whether we sum first on $i$ or on $j$. Eq. (1.13) can also be written in the matrix form, as we will see later.

### 1.5.5 Norm/Length of a Vector

If we let $\mathbf{x}$ and $\mathbf{y}$ be two $n$-dimensional vectors, then their *dot product* is defined as

$$(\mathbf{x} \bullet \mathbf{y}) = \mathbf{x}^T \mathbf{y} = \sum_{i=1}^{n} x_i y_i \tag{1.14}$$

Thus, the dot product is a sum of the product of corresponding elements of the vectors $\mathbf{x}$ and $\mathbf{y}$. Two vectors are said to be *orthogonal* (*normal*) if their dot product is 0; that is, $\mathbf{x}$ and $\mathbf{y}$ are orthogonal if $(\mathbf{x} \bullet \mathbf{y}) = 0$. If the vectors are not orthogonal, the angle between them can be calculated from the definition of the dot product:

$$(\mathbf{x} \bullet \mathbf{y}) = \|\mathbf{x}\| \, \|\mathbf{y}\| \cos\theta, \tag{1.15}$$

where $\theta$ is the angle between vectors $\mathbf{x}$ and $\mathbf{y}$, and $\|\mathbf{x}\|$ represents the *length of vector* $\mathbf{x}$ (also called the *norm of the vector*). The length of vector $\mathbf{x}$ is defined as the square root of the sum of squares of the components:

$$\|\mathbf{x}\| = \sqrt{\sum_{i=1}^{n} x_i^2} = \sqrt{(\mathbf{x} \bullet \mathbf{x})} \tag{1.16}$$

The double sum of Eq. (1.13) can be written in the matrix form as follows:

$$c = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j = \sum_{i=1}^{n} x_i \left( \sum_{j=1}^{n} a_{ij} x_j \right) = \mathbf{x}^T \mathbf{A} \mathbf{x} \tag{1.17}$$

Since $\mathbf{Ax}$ represents a vector, the triple product of Eq. (1.17) is also written as a dot product:

$$c = \mathbf{x}^{\mathrm{T}} \mathbf{A} \mathbf{x} = (\mathbf{x} \bullet \mathbf{A} \mathbf{x}) \tag{1.18}$$

### 1.5.6 Functions of Several Variables

Just as a function of a single variable is represented as $f(x)$, a function of $n$ independent variables $x_1, x_2, \ldots, x_n$ is written as

$$f(\mathbf{x}) = f(x_1, x_2, \ldots, x_n) \tag{1.19}$$

We deal with many functions of vector variables. To distinguish between functions, subscripts are used. Thus, the $i$th function is written as

$$g_i(\mathbf{x}) = g_i(x_1, x_2, \ldots, x_n) \tag{1.20}$$

If there are $m$ functions $g_i(\mathbf{x})$, $i = 1$ to $m$, these are represented in the vector form

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \vdots \\ g_m(\mathbf{x}) \end{bmatrix} = [g_1(\mathbf{x})\, g_2(\mathbf{x}) \ldots g_m(\mathbf{x})]^T \tag{1.21}$$

Throughout the text it is *assumed* that all functions are *continuous* and at least *twice continuously differentiable*. A function $f(\mathbf{x})$ of $n$ variables is *continuous* at a point $\mathbf{x}^*$ if, for any $\varepsilon > 0$, there is a $\delta > 0$ such that

$$|f(\mathbf{x}) - f(\mathbf{x}^*)| < \varepsilon \tag{1.22}$$

whenever $\|\mathbf{x} - \mathbf{x}^*\| < \delta$. Thus, for all points $\mathbf{x}$ in a small neighborhood of point $\mathbf{x}^*$, a change in the function value from $\mathbf{x}^*$ to $\mathbf{x}$ is small when the function is continuous. A continuous function need not be differentiable. *Twice-continuous differentiability* of a function implies not only that it is differentiable two times, but also that its second derivative is continuous.

Fig. 1.5a,b shows continuous and discontinuous functions. The function in Fig. 1.5a is differentiable everywhere, whereas the function in Fig. 1.5b is not differentiable at points $x_1$, $x_2$, and $x_3$. Fig. 1.5c is an example in which $f$ is not a function because it has infinite values at $x_1$. Fig. 1.5d is an example of a discontinuous function. As examples, functions $f(x) = x^3$

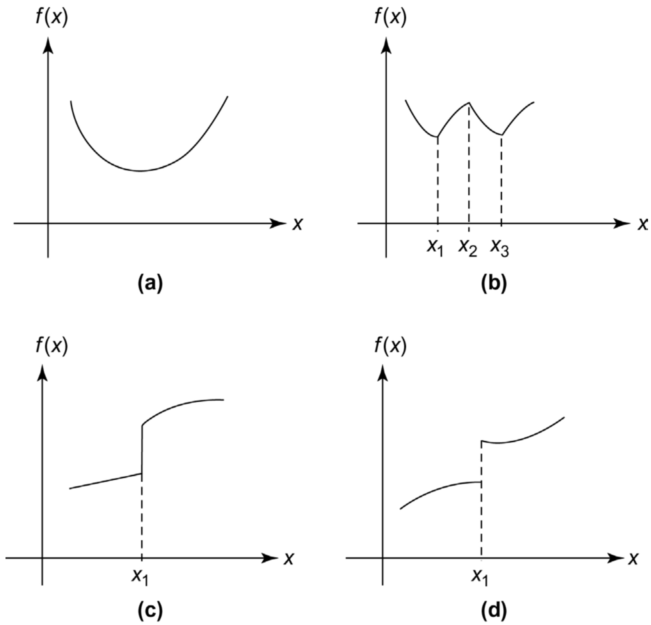


FIGURE 1.5 **Continuous and discontinuous functions.** (a) and (b) Continuous functions; (c) not a function; and (d) discontinuous function.

and $f(x) = \sin x$ are continuous everywhere and are also continuously differentiable. However, function $f(x) = |x|$ is continuous everywhere but not differentiable at $x = 0$.

## 1.5.7 Partial Derivatives of Functions

Often in this text we must calculate derivatives of functions of several variables. Here we introduce some of the basic notations used to represent the partial derivatives of functions of several variables.

### *First Partial Derivatives*

For a function $f(\mathbf{x})$ of $n$ variables, the first partial derivatives are written as

$$\frac{\partial f(\mathbf{x})}{\partial x_i};\ i = 1 \text{ to } n \tag{1.23}$$

The $n$ partial derivatives in Eq. (1.23) are usually arranged in a column vector known as the *gradient* of the function $f(\mathbf{x})$. The gradient is written as $\partial f/\partial \mathbf{x}$ or $\nabla f(\mathbf{x})$. Therefore,

$$\nabla f(\mathbf{x}) = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} [\partial f(\mathbf{x})]/\partial x_1 \\ [\partial f(\mathbf{x})]/\partial x_2 \\ \vdots \\ [\partial f(\mathbf{x})]/\partial x_n \end{bmatrix} \tag{1.24}$$

Note that each component of the gradient in Eqs. (1.23) or (1.24) is a function of vector **x**.

### *Second Partial Derivatives*

Each component of the gradient vector in Eq. (1.24) can be differentiated again with respect to a variable to obtain the second partial derivatives for the function $f(\mathbf{x})$:

$$\frac{\partial^2 f(\boldsymbol{x})}{\partial x_i \, \partial x_j}; \; i, j = 1 \text{ to } n \tag{1.25}$$

We see that there are $n^2$ partial derivatives in Eq. (1.25). These can be arranged in a matrix known as the *Hessian matrix*, written as $\mathbf{H}(\mathbf{x})$, or simply the matrix of second partial derivatives of $f(\mathbf{x})$, written as $\nabla^2 f(\mathbf{x})$:

$$\mathbf{H}(\mathbf{x}) = \nabla^2 f(\mathbf{x}) = \left[ \frac{\partial^2 f(\mathbf{x})}{\partial x_i \, \partial x_j} \right]_{n \times n} \tag{1.26}$$

Note that if $f(\mathbf{x})$ is continuously differentiable two times, then Hessian matrix $\mathbf{H}(\mathbf{x})$ in Eq. (1.26) is *symmetric*.

### *Partial Derivatives of Vector Functions*

On several occasions we must differentiate a vector function of $n$ variables, such as the vector $\mathbf{g}(\mathbf{x})$ in Eq. (1.21), with respect to the $n$ variables in vector $\mathbf{x}$. Differentiation of each component of the vector $\mathbf{g}(\mathbf{x})$ results in a gradient vector, such as $\nabla g_i(\mathbf{x})$. Each of these gradients is an $n$-dimensional vector. They can be arranged as columns of a matrix of dimension $n \times m$, referred to as the gradient matrix of $\mathbf{g}(\mathbf{x})$. This is written as

$$\nabla \boldsymbol{g}(\boldsymbol{x}) = \frac{\partial \boldsymbol{g}(\boldsymbol{x})}{\partial \boldsymbol{x}} = \left[ \nabla g_1(\boldsymbol{x}) \;\; \nabla g_2(\boldsymbol{x}) \;\; \ldots \;\; \nabla g_m(\boldsymbol{x}) \right]_{n \times m} \tag{1.27}$$

This gradient matrix is usually written as matrix **A**:

$$\boldsymbol{A} = \left[ a_{ij} \right]_{n \times m}; \; a_{ij} = \frac{\partial g_j}{\partial x_i}; \; i = 1 \text{ to } n; \; j = 1 \text{ to } m \tag{1.28}$$

## 1.5.8 US–British Versus SI Units

The formulation of the design problem and the methods of optimization do not depend on the units of measure used. Thus, it does not matter which units are used to formulate the problem. However, the final form of some of the analytical expressions for the problem does depend on the units used. In the text, we use both US–British and SI units in examples and exercises. Readers unfamiliar with either system should not feel at a disadvantage when reading and understanding the material since it is simple to switch from one system to the other. To facilitate the conversion from US–British to SI units or vice versa, Table 1.1 gives

**TABLE 1.1** Conversion Factors for US–British and SI Units

| To convert from US–British | To SI units | Multiply by |
|---|---|---|
| **Acceleration** | | |
| Foot/second$^2$ (ft./s$^2$) | Meter/second$^2$ (m/s$^2$) | 0.3048* |
| Inch/second$^2$ (in./s$^2$) | Meter/second$^2$ (m/s$^2$) | 0.0254* |
| **Area** | | |
| Foot$^2$ (ft.$^2$) | Meter$^2$ (m$^2$) | 0.09290304* |
| Inch$^2$ (in.$^2$) | Meter$^2$ (m$^2$) | 6.4516E–04* |
| **Bending moment or torque** | | |
| Pound force inch (lbf·in.) | Newton meter (N·m) | 0.1129848 |
| Pound force foot (lbf·ft.) | Newton meter (N·m) | 1.355818 |
| **Density** | | |
| Pound mass/inch$^3$ (lbm/in.$^3$) | Kilogram/meter$^3$ (kg/m$^3$) | 27,679.90 |
| Pound mass/foot$^3$ (lbm/ft.$^3$) | Kilogram/meter$^3$ (kg/m$^3$) | 16.01846 |
| **Energy or work** | | |
| British thermal unit (BTU) | Joule (J) | 1055.056 |
| Foot pound force (ft.·lbf) | Joule (J) | 1.355818 |
| Kilowatt-hour (KWh) | Joule (J) | 3,600,000* |
| **Force** | | |
| Kip (1000 lbf) | Newton (N) | 4448.222 |
| Pound force (lbf) | Newton (N) | 4.448222 |
| **Length** | | |
| Foot (ft.) | Meter (m) | 0.3048* |
| Inch (in.) | Meter (m) | 0.0254* |
| Inch (in.) | Micron (μ); micrometer (μm) | 25,400* |
| Mile (mi), US statute | Meter (m) | 1609.344 |
| Mile (mi), International, nautical | Meter (m) | 1852* |
| **Mass** | | |
| Pound mass (lbm) | Kilogram (kg) | 0.4535924 |
| Ounce | Grams | 28.3495 |
| Slug (lbf·s$^2$ft.) | Kilogram (kg) | 14.5939 |
| Ton (short, 2000 lbm) | Kilogram (kg) | 907.1847 |
| Ton (long, 2240 lbm) | Kilogram (kg) | 1016.047 |
| Tonne (t, metric ton) | Kilogram (kg) | 1000* |
| **Power** | | |
| Foot pound/minute (ft.·lbf/min) | Watt (W) | 0.02259697 |
| Horsepower (550 ft. lbf/s) | Watt (W) | 745.6999 |

**TABLE 1.1** Conversion Factors for US–British and SI Units (*cont.*)

| To convert from US–British | To SI units | Multiply by |
|---|---|---|
| **Pressure or stress** | | |
| Atmosphere (std) (14.7 lbf/in.$^2$) | Newton/meter$^2$ (N/m$^2$ or Pa) | 101,325* |
| One bar (b) | Newton/meter$^2$ (N/m$^2$ or Pa) | 100,000* |
| Pound/foot$^2$ (lbf/ft.$^2$) | Newton/meter$^2$ (N/m$^2$ or Pa) | 47.88026 |
| Pound/inch$^2$ (lbf/in.$^2$ or psi) | Newton/meter$^2$ (N/m$^2$ or Pa) | 6894.757 |
| **Velocity** | | |
| Foot/minute (ft./min) | Meter/second (m/s) | 0.00508* |
| Foot/second (ft./s) | Meter/second (m/s) | 0.3048* |
| Knot (nautical mi/h), international | Meter/second (m/s) | 0.5144444 |
| Mile/hour (mi/h), international | Meter/second (m/s) | 0.44704* |
| Mile/hour (mi/h), international | Kilometer/hour (km/h) | 1.609344* |
| Mile/second (mi/s), international | Kilometer/second (km/s) | 1.609344* |
| **Volume** | | |
| Foot$^3$ (ft.$^3$) | Meter$^3$ (m$^3$) | 0.02831685 |
| Inch$^3$ (in.$^3$) | Meter$^3$ (m$^3$) | 1.638706E–05 |
| Gallon (Canadian liquid) | Meter$^3$ (m$^3$) | 0.004546090 |
| Gallon (UK liquid) | Meter$^3$ (m$^3$) | 0.004546092 |
| Gallon (UK liquid) | Liter (L) | 4.546092 |
| Gallon (US dry) | Meter$^3$ (m$^3$) | 0.004404884 |
| Gallon (US liquid) | Meter$^3$ (m$^3$) | 0.003785412 |
| Gallon (US liquid) | Liter (L) | 3.785412 |
| One liter (L) | Meter$^3$ (m$^3$) | 0.001* |
| One liter (L) | Centimeter$^3$ (cm$^3$) | 1000* |
| One milliliter (mL) | Centimeter$^3$ (cm$^3$) | 1* |
| Ounce (UK fluid) | Meter$^3$ (m$^3$) | 2.841307E–05 |
| Ounce (US fluid) | Meter$^3$ (m$^3$) | 2.957353E–05 |
| Ounce (US fluid) | Liter (L) | 2.957353E–02 |
| Ounce (US fluid) | Milliliter (mL) | 29.57353 |
| Pint (US dry) | Meter$^3$ (m$^3$) | 5.506105E–04 |
| Pint (US liquid) | Liter (L) | 4.731765E–01 |
| Pint (US liquid) | Meter$^3$ (m$^3$) | 4.731765E–04 |
| Quart (US dry) | Meter$^3$ (m$^3$) | 0.001101221 |
| Quart (US liquid) | Meter$^3$ (m$^3$) | 9.463529E–04 |

* *Exact conversion factor.*

conversion factors for the most commonly used quantities. For a complete list of conversion factors, consult the IEEE/ASTM (2010) publication.

## Reference

IEEE/ASTM, 2010. American National Standard for Metric Practice. SI 10-2010. The Institute of Electrical and Electronics Engineers/American Society for Testing of Materials, New York.